PATENT APPLICATION

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE
BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES

In re application of

Docket No: STL000023US1/A851+ RECEIVED

BAER, William J., et al.

MAR 2 6 2003

Appln. No.: 09/489,605

Group Art Unit: 2171   Technology Center 2100

Confirmation No.: 6038

Examiner: Nguyen, Cam Linh T

Filed: January 21, 2000

For: PROVIDING A FUNCTIONAL LAYER FOR FACILITATING CREATION AND MANIPULATION OF COMPILATIONS OF CONTENT

SUBMISSION OF APPELLANT'S BRIEF ON APPEAL

Commissioner for Patents
Washington, D.C. 20231

Sir:

Submitted herewith please find an original and two copies of Appellant's Brief on Appeal. A check for the statutory fee of $320.00 is attached. The USPTO is directed and authorized to charge all required fees, except for the Issue Fee and the Publication Fee, to Deposit Account No. 19-4880. Please also credit any overpayments to said Deposit Account. A duplicate copy of this paper is attached.

Respectfully submitted,

J. Warren Lytle, Jr.
Registration No. 39,283

SUGHRUE MION, PLLC
Telephone: (202) 293-7060
Facsimile: (202) 293-7860

WASHINGTON OFFICE

23373

PATENT TRADEMARK OFFICE

Date: March 24, 2003

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE
BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES

In re application of	Docket No: STL000023US1/A8514
BAER, William J., et al.	**RECEIVED**
	MAR 2 6 2003
Appln. No.: 09/489,605	Group Art Unit: 2171
	Technology Center-2100
Confirmation No.: 6038	Examiner: Nguyen, Cam Linh T

Filed: January 21, 2000

For: PROVIDING A FUNCTIONAL LAYER FOR FACILITATING CREATION AND
MANIPULATION OF COMPILATIONS OF CONTENT

APPELLANTS' BRIEF ON APPEAL UNDER 37 C.F.R. § 1.192

Commissioner for Patents
Washington, D.C. 20231

Sir:

In accordance with the provisions of 37 C.F.R. § 1.192, Appellant submits the following:

I.   REAL PARTY IN INTEREST

The real party in interest in this appeal is International Business Machines Corporation

("IBM") of Armonk, New York, the assignee.

II.   RELATED APPEALS AND INTERFERENCES

There are no other appeals or interferences known to Appellant, Appellant's legal

representative, or the assignee that will directly affect or be directly affected by, or have a

bearing on, the Board's decision in this appeal.

III.   STATUS OF CLAIMS

Claims 1 - 18 are the claims pending in the application and are the subject of this appeal.

A copy of the claims on appeal are set forth in an attached Appendix.

IV.   STATUS OF AMENDMENTS

All Amendments submitted have been entered. A Response Under 37 C.F.R. § 1.116

was filed on November 25, 2002, in response to a Final Office Action (Paper No. 12), and the

Examiner indicates in the Advisory Action (Paper No. 14) that the Response was considered.

V.   SUMMARY OF THE INVENTION

The application on appeal relates to a content management system for creating

compilations of content entities stored in a data repository. The compilation defines a group of

hierarchically related content entities. The content entity at each level of the hierarchy, except

for at the lowest level, is defined by a container. At the lowest level of the hierarchy the content

entity is multimedia content data. The content can encompass many forms of multimedia data.

In the case of textural content, the hierarchical structure of the data corresponds to the structure

of a book and would include a book container, volume containers, chapter containers and

subsections. The subsections are not containers, but rather are digital content data, since they are

located at the leaf level of the hierarchy.

The content entities and data associated with them preferably are stored in a digital

library that includes a search support capability. A World-Wide Web based user interface

presents a user with a plurality of selectable objects in which each object represents a subset of

the hierarchical content entities stored in the digital library. The user selects one or more of the

2

objects for inclusion in the compilation. An example of such a compilation is a custom textbook.

The system creates a custom book outline (CBO) that defines the compilation as a list or outline

of the content entities that were selected. The CBO includes identifiers of those content entities,

ordered in the CBO according to the user's specification. An example of a CBO is shown at pg.

61. The CBO includes identifiers of content entities held within the digital library. The structure

and order of the content entities selected for inclusion in the compilation can be created and

manipulated by creating the CBO. The CBO can be manipulated to manipulate the compilation.

Fig. 6 in the application shows a selection path for creating a compilation of content. As

shown in this path a user web interface 22, which preferably runs within a standard web browser,

communicates through a web server 26 to an application layer 28. The application layer 28 can

include sets of PERL applications that interface with the user interface 22 to handle certain

transactions such as performing CBO manipulation. Layer 28 also forwards commands to an

application programming interface (API) layer 30 that interfaces with a digital library 20 to

communicate actions requested by the user to the digital library. See pg. 59.

The application layer 28 and the API layer 30 interoperate to execute a group of functions

that enable creation, submission and request processing of a CBO. As described at page 63 of

the application, the application layer 28 includes functional modules written in the PERL script

language that correspond to C++ scripts in the API layer 30. Various CBO functions

implemented by the application layer and API layer are listed on page 64. They include, for

example, ECBOCreate, ECBOAddContent, ECBORemoveContent, ECBOMoveContent, etc.

Through these function modules a user can create and manipulate the compilation of content.

The application layer 28 interfaces the user interface 22 and the API layer 30 interfaces

with the digital library 20. The application layer 28 interfaces with the user web interface 22

through a network by way of a web server 26.

Each content entity can be associated with a price. When the compilation of content

entities is formed, a price for the compilation, such as the custom book, is calculated based on

the prices associated with the selected content entities. See function modules ECBOGet,

discussed at page 72; ECBOMoveContent and ECBORelocateContent, discussed at pg. 86; and

EAdminPopulatePageAndPrice discussed at pg. 91, for example.

VI.   ISSUES

The sole issue on appeal is whether claims 1-18 are anticipated, under 35 U.S.C. §

102(e), by U.S. Patent No. 6,199,082 to Ferrel et al. ("Ferrel").

VII.   GROUPING OF CLAIMS

The claims do not stand or fall together and arguments for patentability of each group of

claims, identified below, are set forth in this brief.

Group I: Claims 1 through 15, each of which stand or fall together.

Group II: Claims 16 through 18, each of which stand or fall together.

VIII. ARGUMENTS

The Prior Art: Ferrel

Ferrel relates to a multimedia publishing system in which the design and content of a

publication, or title, are stored separately to facilitate transmitting a variety of changing content

4

to customers. See Abstract. The content and design are stored as separate objects in a public distribution site so that many different pieces of content can be viewed with the same appearance. Col. 8, lines 15-18. A designer, using project designer software at the publisher's site creates a particular page layout with which many pieces of content can be viewed. Col. 8, lines 21-25.

Fig. 2 shows the components of the Ferrel multimedia publishing (MP) system. A publisher 102 uses a publisher workstation 180 that has a designer software environment 194, indicated by the bracket in Fig. 2, to create and publish title layouts and content. Col. 11, lines 50 – 52. Ferrel discloses that the designer software environment 194 is a development environment that includes several software components, such as a project editor 184, a page editor 186, a style sheet editor 187, a search object editor 189, and a word processor 188. A publisher uses the designer software environment 194 to create and publish title layouts and content. The layouts and content, once created by the publisher, are released to a host data center within network 122, in which they are stored.

Referring to Fig. 3, the host data center 242 includes MP system servers (MPS servers) 246a though 246c that store the released layouts and content. Col. 11, lines 55-59. Customers, using customer workstations 160, 162 and 164, shown in Fig.1, that include a viewer 202, shown in Fig. 2, can acquire the content and layout either from local storage within the customer workstation or from the host data center's publication storage 120 in the network 122. See col. 13, lines 50-162. Ferrel et al. does not disclose, or even suggest, any interaction between the customer workstation 182 and the publisher workstation 102 that holds the designer software

5

components 194. The customer workstation interacts only with the host data center's publication

storage 120 in network 122.

Group I (Claims 1-15): Ferrel Does Not Disclose Function Modules for Creating or Manipulating a Compilation of Content, that Both Interface Over a Network with a User Interface and that Interface with a Data Repository Containing Content Entities From Which the Compilation is Created.

Claim 1 is directed to a method for facilitating the creation and manipulation of

compilations of content. The claim requires a functional layer for interfacing over a network

with a user interface. It further requires that the functional layer also interface with a data

repository that contains a plurality of content entities. The functional layer includes a plurality of

function modules each of which executes a function that pertains to the creation or manipulation

of a compilation of content. The content forming the compilation is selected from the plurality

of content entities held in the data repository. The function modules receive requests from the

user interface and return information concerning the compilation of content.

Ferrel does not disclose, or even suggest, a functional layer that interfaces with both a

user interface over a network and a data repository that contains a plurality of content entities,

and that includes functional modules that create and manipulate a compilation of content.

In the final Office Action the Examiner takes the position that the publisher workstation

102, shown in Fig. 2 of Ferrel, is a functional layer that contains a plurality of function modules

(i.e., designer software components 184, 186-190 and 192-193). The Examiner further asserts

that the publisher's designer software components interface with the customer workstation 182,

160, because "[t]hese functional modules communicate with user interface 182 by a network

122." See pg. 2 of the final Office Action. The Examiner further states, at page 4 of the final

Office Action, that "referring to Fig. 2 clearly a user 182 interfaces with a workstation 180," and

"[t]he publisher 102 receives the request from user 182, and returns information regarding the

compilation of content." It is respectfully submitted that the Examiner misapprehends the

system disclosed by Ferrel.

Nowhere does Ferrel disclose that the publisher 102 receives a request from customer

workstation 182. Any requests from customer workstation 182 are sent only to the host data

center 242 which resides in network 122, because the content and layout objects released by the

publisher are stored in that host data center, not at the publisher workstation 102.

The publisher creates the content and layout for titles at the publisher's workstation 102

using the designer software components without receiving any requests from customer

workstation 182. Ferrel discloses that publishers, working on their own workstation 180, create a

project using the designer software 194 (i.e., page editor 186, style sheet editor 187, search

object editor 189, etc.). See col. 11, lines 49 – 53 and steps 324-330 in Fig. 5. Once that project

is created the publisher, in step 332, releases it by storing the design layout and content for that

project on the host data center 242 (see Fig. 3) which is part of network 122 (see Fig. 2). Ferrel

specifically discloses, at col. 11, lines 55 – 58, that "title layout and content are stored in a

network 122 that include a server for hosting on-line applications." Those servers (246a, 246b

and 246c) are shown as part of the host data center in Fig. 3, which is in network 122.

Ferrel does not show, or even suggest, that the designer software modules 184, 186 - 190, and 192 – 193, receive a request, either directly or indirectly, from the customer workstation 182. Rather, Ferrel discloses only that the customer workstation interacts with the network 122 (not the publisher's workstation) to receive publications that have been released by a publisher.

Ferrel describes the interactions between publishers and customers with respect to Fig. 5, at cols. 18 – 19. Fig. 5 illustrates a process whereby a publisher, working on workstation 180 creates a publication using designer software 194 that includes the software modules, asserted in the Office Action to correspond with the claimed function modules. See col. 11, lines 49 - 53. Referring to Fig. 5, a publisher creates a project that includes layout objects and content objects, in steps 324 through 330, and then releases those projects by storing them on the host data center 242, in step 332. Ferrel specifically discloses, at col. 11, lines 55 – 58, that "title layout and content are stored in a network 122 that include a server for hosting on-line applications." Those servers are shown as part of the host data center shown in Fig. 3. Once the titles are published to the host data center, the customer workstation interacts only with the network 122 to access those released titles stored in the host data center 242, and not with the publisher or the designer software modules. Fig. 5 illustrates this in step 334 in which a customer uses a viewer 202 that is part of a customer workstation shown in Fig. 2, to read and page through the released titles.

Accordingly, it is respectfully submitted that Ferrel does not anticipate the claim 1. Claims 6 and 11 recite similar limitations, and it is respectfully submitted that they are not anticipated for at least the same reasons.

APPELLANTS' BRIEF ON APPEAL
UNDER 37 C.F.R. § 1.192
U.S. Appln. No.: 09/489,605

Claims 2-5, 7-10 and 12-15 contain by reference all the limitations of one of claims 1, 6

and 11, and hence, are not anticipated for at least the same reasons.

Group II (Claims 16-18): Since Ferrel Does Not Disclose Calculating a Price for a Compilation of Content, Ferrel Does Not Anticipate Claims 16-18.

Claim 16, for example, requires that at least one of the function modules calculates a

price for the compilation of content. Ferrel neither teaches nor suggests that limitation.

The Examiner refers to two portions of Ferrel as disclosing this feature. The first is at

col. 3, lines 18 – 20, which merely discloses, "Content providers might also wish to create both

'full' and 'lite' versions of their titles, where the 'lite' versions contains less content for a smaller

price." The second disclosure is at col. 29, lines 103, which states, "A title may have a price

associated with it, which is set using the MSN sysop tools."

The MPEP, §2131 (8th ed. 2001) states the test for anticipation. "A claim is anticipated

only if each and every element as set forth in the claim is found, either expressly or inherently

described, in a single prior art reference." *Verdegaal Bros. v. Union Oil Co. of California*, 814

F.2d 628, 631, 2 USPQ2d 1051, 1053 (Fed. Cir. 1987). "The identical invention must be shown

in as complete detail as is contained in the ... claim." *Richardson v. Suzuki Motor Co.*, 868 F.2d

1226, 1236, 9 USPQ2d 1913, 1920 (Fed. Cir. 1989).

Ferrel does not disclose any function module that calculates a price for the compilation of

content as required by claims 16-18. Ferrel does not disclose calculating a price, but merely

indicates that a price for a title can be set. There is not even a suggestion of calculating a price,

9

much less that the price is calculated by one of Ferrels' alleged function modules, namely, one of

the publisher's designer software components 194. Accordingly, since Ferrel does not disclose

all the limitations of claims 16-18 in as complete detail as recited in the claims, Ferrel can not

anticipate those claims.

The present Brief on Appeal is being filed in triplicate. Unless a check is submitted

herewith for the fee required under 37 C.F.R. §1.192(a) and 1.17(c), please charge said fee to

Deposit Account No. 19-4880.

The USPTO is directed and authorized to charge all required fees, except for the Issue

Fee and the Publication Fee, to Deposit Account No. 19-4880. Please also credit any

overpayments to said Deposit Account.

Respectfully submitted,

SUGHRUE MION, PLLC
Telephone: (202) 293-7060
Facsimile: (202) 293-7860

J. Warren Lytle, Jr.
Registration No. 39,283

WASHINGTON OFFICE

23373

PATENT TRADEMARK OFFICE

Date: March 24, 2003

10

APPENDIX

CLAIMS 1-18 ON APPEAL:

1   1. (Twice Amended) A method for facilitating creation and manipulation of

2   compilations of content, comprising the step of:

3   providing a functional layer for interfacing over a network with a user interface and

4   interfacing with a data repository containing a plurality of content entities, the functional layer

5   comprising a plurality of function modules, each module for executing a function pertaining to

6   the creation or manipulation of a compilation of content selected from the plurality of content

7   entities, wherein the plurality of function modules receives requests from the user interface and

8   returns information concerning the compilation of content.

1   2. (Amended) The method of claim 1, wherein one of the function modules creates a list

2   of content entity identifiers defining the content and order of a compilation.

1   3.   The method of claim 2, wherein at least one of the function modules manipulates

2   the list to redefine the content or order of the compilation.

1   4. (Amended) The method of claim 1, wherein the compilation is hierarchically

2   structured and wherein one of the function modules creates an outline of containers and content

3   entity identifiers defining the content and hierarchical structure of a compilation.

1   5.   The method of claim 4, wherein at least one of the function modules manipulates

2   the outline to redefine the content or structure of the compilation.

1   6. (Amended) A program storage device readable by a machine, tangibly embodying a

2   program of instructions executable by the machine to perform method steps for facilitating

3   creation and manipulation of compilations of content, comprising the step of:

4   providing a functional layer for interfacing over a network with a user interface and

5   interfacing with a data repository containing a plurality of content entities, the functional layer

6   comprising a plurality of function modules, each module for executing a function pertaining to

7   the creation or manipulation of a compilation of content selected from the plurality of content

8   entities, wherein the plurality of function modules receives requests from the user interface and

9   returns information concerning the compilation of content.

1   7. (Amended) The program storage device of claim 6, wherein one of the function

2   modules creates a list of content entity identifiers defining the content and order of a

3   compilation.

1   8. (Amended) The program storage device of claim 7, wherein at least one of the

2   function modules manipulates the list to redefine the content or order of the compilation.

1   9. (Amended) The program storage device of claim 6, wherein the compilation is

2   hierarchically structured and wherein one of the function modules creates an outline of

3   containers and content entity identifiers defining the content and hierarchical structure of a

4   compilation.

1   10. (Amended) The program storage device of claim 9, wherein at least one of the

2   function modules manipulates the outline to redefine the content or structure of the compilation.

1   11. (Amended)   A system for facilitating creation and manipulation of compilations

2   of content, comprising:

3   a functional layer for interfacing over a network with a user interface and interfacing with

4   a data repository containing a plurality of content entities; and

5   a plurality of function modules in the functional layer, each module including means for

6   executing a function pertaining to the creation or manipulation of a compilation of content

7   selected from the plurality of content entities, wherein the plurality of function modules receives

8   requests from the user interface and returns information concerning the compilation of content.

1   12. (Amended) The system of claim 11, wherein one of the function modules further

2   comprises means for creating a list of content entity identifiers defining the content and order of

3   a compilation.

1   13.   The system of claim 12, wherein at least one of the function modules further

2   comprises means for manipulating the list to redefine the content or order of the compilation.

1   14.(Amended) The system of claim 11, wherein the compilation is hierarchically

2   structured and wherein one of the function modules further comprises means for creating an

13

3   outline of containers and content entity identifiers defining the content and hierarchical structure

4   of a compilation.

1   15.   The system of claim 14, wherein at least one of the function modules further

2   comprises means for manipulating the outline to redefine the content or structure of the

3   compilation.

1   16.   The method of claim 1, wherein at least one of the function modules calculates a

2   price for the compilation of content.

1   17.   The program storage device of claim 6, wherein at least one of the function

2   modules calculates a price for the compilation of content.

1   18.   The system of claim 11, wherein at least one of the function modules calculates a

2   price for the compilation of content.